AMENDMENTS TO THE CLAIMS:

This listing of claims replaces all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1. (Previously Presented) A method of generating a logic design for use in designing an integrated circuit (IC), comprising:

embedding a computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of declarations;

wherein the two-dimensional schematic representation includes a set of Register Transfer Diagrams (RTD).

2. (Previously presented) The method of claim 1, further comprising generating the computer instruction.

3. (Currently Amended) The method of claim 2, further comprising importing the computer instruction from memory.

4. (Currently Amended) The method of claim 3, wherein the computer instruction is devoid of entries to a ~~selectivity~~ sensitivity list.

5. (Currently Amended) The method of claim 4, further comprising notifying a designer when capturing data using the computer instruction violates a set of design capture rules.

6. (Currently Amended) The method of claim 1, further comprising using a set of abstractions to represent hierarchical levels of the logic design.

7. (Original) The method of claim 1, further comprising generating C++ from the unified database.

8. (Previously Presented) The method of claim 7, further comprising generating Verilog from the unified database.

9. (Cancelled)

10. (Original) The method of claim 1, further comprising generating synthesizable Verilog from the unified database.

11. (Previously Presented) An article comprising a machine-readable medium which stores executable instructions to generate a logic design for use in designing an integrated circuit (IC), the instructions causing a machine to:

embed a computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of declarations;

wherein the two-dimensional schematic representation includes a set of Register Transfer Diagrams (RTD).

12. (Previously Presented) The article of claim 11, further comprising instructions causing a machine to generate the computer instruction.

13. (Currently Amended) The article of claim 12, wherein the computer instruction is devoid of entries to a ~~selectivity~~ sensitivity list.

14. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to import the computer instruction.

15. (Currently Amended) The article of claim 12, further comprising instructions causing a machine to notify a designer when capturing data violates a [[e]] set of design capture rules.

16. (Previously Presented) The article of claim 11, further comprising causing a machine to use a set of abstractions.

17. (Previously Presented) The article of claim 11, further comprising instructions causing a machine to generate C++ from the unified database.

18. (Previously Presented) The article of claim 12, further comprising instructions causing a machine to generate Verilog from the unified database.

19. (Cancelled)

20. (Previously Presented) The article of claim 11, further comprising instructions causing a machine to generate synthesizable Verilog from the unified database.

21. (Previously Presented) An apparatus for generating a logic design for use in designing an integrated circuit (IC), comprising:

a memory that stores executable instructions; and

a processor that executes the instructions to:

embed a computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of declarations;

wherein the two-dimensional schematic representation includes a set of Register Transfer Diagrams (RTD).

22. (Previously Presented) The apparatus of claim 21, further comprising instructions to generate the computer instruction.

23. (Currently Amended) The apparatus of claim 22, wherein the computer instruction is devoid of entries to a sensitivity list.

24. (Previously Presented) The apparatus of claim 23, further comprising instructions to import the computer instruction.

25. (Previously Presented) The apparatus of claim 23, further comprising instructions to notify a designer when capturing data violates a set of design capture rules.

26. (Original) The apparatus of claim 21, further comprising instructions to use a set of abstractions.

27. (Original) The apparatus of claim 21, further comprising instructions to generate C++ from the unified database.

28. (Previously Presented) The apparatus of claim 27, further comprising instructions to generate Verilog from the unified database.

29. (Cancelled)

30. (Original) The apparatus of claim ~~29~~ 21, further comprising instructions to generate synthesizable Verilog from the unified database.

31. (Previously Presented) The method of claim 1, further comprising enabling a user to change the logic design by amending the computer instruction.

32. (Previously Presented) The article of claim 11, further comprising instructions causing a machine to enable a user to change the logic design by amending the computer instruction.

33. (Previously Presented) The apparatus of claim ~~29~~ 21, further comprising instructions to enable a user to change the logic design by amending the computer instruction.